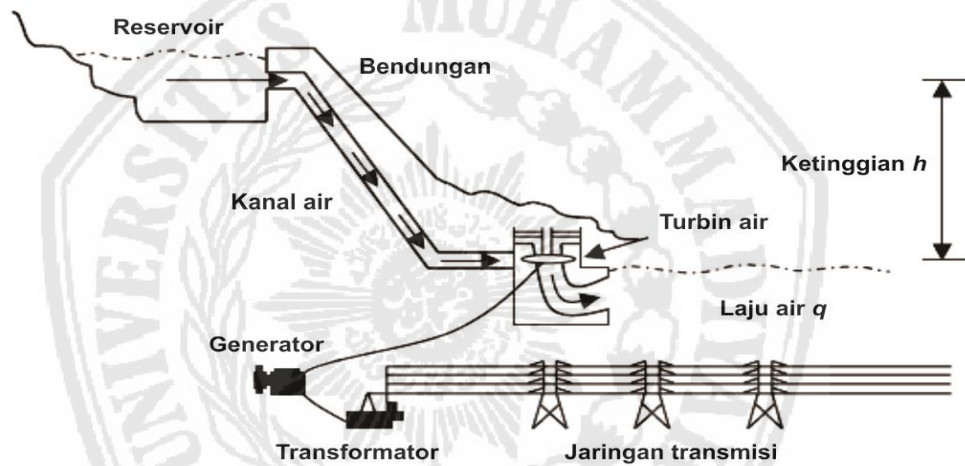


BAB II

TINJAUAN PUSTAKA

2.1 PEMBANGKIT LISTRIK TENAGA MIKRO HIDRO (PLTMH)

Pembangkit Listrik Tenaga Mikrohidro (PLTMH) adalah pembangkit listrik untuk skala kecil yang memanfaatkan aliran air sebagai sumber energi. Aliran air tersebut dialirkan ke pipa dan disalurkan ke turbin. Turbin memutar poros generator untuk menghasilkan energi listrik. Pada prinsipnya PLTMH menggunakan jumlah debit air pada ketinggian tertentu.



Gambar 2.1 Proses alur kerja energi pada PLTMH

2.2 GOVERNOR

Fungsi utama dari sistem pengaturan turbin adalah untuk mengatur kecepatan generator dan karenanya frekuensi dan daya aktif sebagai respon terhadap variasi beban. Mekanisme kontrol kecepatan termasuk peralatan relay, servo motor, tekanan atau perangkat penguatan daya, pegungkit dan hubunfan antara kecepatan dan *gate* yang dikendalikan governor. Kecepatan putaran generator diatur oleh governor yang mengatur input air ke turbin melalui mekanisme kontrol kecepatan.

Kecepatan *permanent droop* (R_p) menentukan jumlah perubahan dalam *output* yang dihasilkan unit sebagai respon terhadap perubahan dalam satuan kecepatan. Kecepatan *permanent droop* dapat dikembangkan dengan menggunakan

posisi gerbang gawang atau dengan menggunakan daya *output* unit. Untuk operasi terisolasi atau kepulauan, penurunan kecepatan permanen dapat membantu menstabilkan kendali kecepatan unit dengan memberikan umpan balik antara yang membatasi perjalanan lebih dari servo pengendali turbin sambil mengendalikan kecepatan unit. Penurunan kecepatan permanen menentukan jumlah partisipasi yang dihasilkan unit ketika menanggapi gangguan dalam frekuensi sistem dalam operasi ketika disinkronisasi ke sistem tenaga yang saling berhubungan.

Temporary droop (R_t) digunakan untuk membatasi overshoot dari servomotor kontrol turbin selama kondisi transien. *Temporary droop* dapat dikembangkan baik menghubungkan *dashpot* dari posisi gerbang gawang ke titik penjumlahan kesalahan governor, atau menambahkan turunan yang disaring dari posisi gerbang gawang ke titik penjumlahan *error* governor.

2.3 PERALATAN KONTROL

Sistem kontrol pada PLTMH pada dasarnya ada sistem kontrol tegangan dan sistem kontrol frekuensi. Sebagai pengontrol tegangan adalah *Automatic Voltage Regulator* (AVR) dan sebagai pengontrol frekuensi adalah FCV dan ELC.

2.3.1 *Flow Control Valve* (FCV)

Flow Control Valve adalah suatu peralatan kontrol untuk mengatur putaran turbin relative konstan untuk berbagai kondisi beban. FCV berguna untuk merespon nilai *error* pada frekuensi yang mana untuk mengontrol *valve* pada turbin untuk mengatur aliran air melalui *penstock*.

a. Pemodelan Turbin Hidrolik

Menurut penelitian dari Amir Kumar Singh yang berjudul “*Modelling and Simulation of Micro Hydro Diesel Hybrid Power System for Localized Power Requirement Using MATLAB/Simulink*” menjelaskan pemodelan turbin hidrolik dan *governor*. Pemodelan dirumuskan dengan

$$Q = G\sqrt{H} \dots\dots\dots (2-1)$$

Dimana :

$Q = \text{turbine flow (m}^3/\text{s)}.$

$G = \text{gate opening (rad)}.$

$H = \text{net head (m)}.$

$$Pm = A_t H (\bar{Q} - \bar{Q}_{nl}) \dots\dots\dots (2-2)$$

Dimana :

$\bar{Q} = \text{Turbine flow (pu)}$

$A_t = \text{Turbine gain}$

$Q_{nl} = \text{No load flow (pu)}$

$$A_t = \frac{1}{(\bar{G}_{\max} - \bar{G}_{\min})} \dots\dots\dots (2-3)$$

Persamaan 2.1 dikembangkan untuk mendapatkan perubahan aliran air di *penstock* sehingga didapat.

$$U = K_u G \sqrt{H} \dots\dots\dots (2-4)$$

Dimana :

$U = \text{Velocity of water}$

$K_u = \text{Proportional constant}$

Setelah kecepatan air di *penstock* ditentukan, hubungan laju aliran, pada *head* bias ditentukan dengan persamaan sebagai berikut.

$$Q = AU \dots\dots\dots (2-5)$$

Percepatan aliran dalam *penstock* dijelaskan pada persamaan (2-6)

$$\frac{du}{dt} = \frac{-ag}{L} (H - H_0) \dots\dots\dots (2-6)$$

Dimana :

$ag = \text{acceleration due gravity}$

$L = \text{length of penstock}$

Normalisasi persamaan (2-4) tentang nilai-nilai dinilai

$$\bar{H} = \left(\frac{\bar{H}}{\bar{G}}\right)^2 \dots\dots\dots (2-7)$$

Dengan memberikan persamaan

$$T_w = \frac{LQ_{base}}{agAH_{base}} \dots\dots\dots (2-8)$$

Sehingga didapatkan persamaan sebagai berikut.

Dimana:

T_w = *water starting time* (s)

Q_{base} = *turbine flow* (m³/s)

H_{base} = *head turbine* (m).

Output tenaga mesin diberikan oleh

Dengan memberikan

Dimana :

P_L = Power Loss

U_{NL} = No Load speed

Persamaan 2.1, 2.3, 2.7 dan 2.9 dapat dikombinasikan untuk menghasilkan karakteristik dinamik umum turbin hidrolik.

b. Pemodelan *hydro-electric servo system*

Dalam model *hydro-electric servo system* digunakan sebagai aktuator untuk membuka katup dengan mendapatkan sinyal *error* sebagai nilai input, aktuator disini adalah motor servo. Kontrol umpan balik dari posisi sudut ke aktuator digeser oleh katup untuk meningkatkan atau mengurangi pembukaan gerbang untuk menstabilkan sistem dengan mempertahankan kecepatan konstan. *Hydro-electric servo system* digunakan untuk mengontrol bukaan pada katup berdasarkan sinyal dari kontroler.

Ada dua lilitan yang ditempatkan pada stator yang dikenal sebagai fase kontrol dan referensi. Tegangan yang diaplikasikan pada fase V1 dan V2 ini harus 90° dari fase dengan besaran yang tidak sama. Karena pergeseran fasa ini, medan magnet berputar. Tegangan referensi dijaga konstan dan tegangan fase kontrol digerakkan dengan menggerakkan sinyal kesalahan. Daya keluaran mekanik motor

servo bervariasi dari 2 watt sampai beberapa ratus watt. Torsi motor adalah fungsi sinyal kecepatan dan kesalahan. Berikut persamaannya.

$$T_m = f(\dot{\theta}, e) \dots\dots\dots (2-9)$$

Torsi dari motor servo dapat bervariasi dengan merubah besarnya tegangan fase kontrol. Dengan demikian, arah rotasi dapat diubah dengan mengubah polaritas kontrol tegangan.

Torsi (persamaan 2-9) motor servo dapat diperluas dengan menggunakan deret Taylor seperti yang ditunjukkan pada persamaan (2-10).

$$T_m = (t_a(0) + \frac{dt_a}{de}(e(t) - e(0) + \dots + \frac{dt_a}{d\dot{\theta}}(\dot{\theta}(t) - \dot{\theta}(0) + \dots \quad (2-10)$$

Dengan mengabaikan persyaratan pada orde yang lebih tinggi dan mengingat kondisi awal nol. Persamaan (2-11) dapat ditulis sebagai berikut.

$$T_m = K_e(t) - f\dot{\theta}(t) \dots\dots\dots (2-11)$$

Dimana $K = \frac{dT_m}{de}$ dan $f = -\frac{dT_m}{d\dot{\theta}}$

Kita tahu hubungan mekanis motor adalah,

$$T_m = J\ddot{\theta} + B\dot{\theta} \dots\dots\dots (2-12)$$

Dimana, J dan B adalah koefisien gesekan dan momen inersia masing-masing.

Dari persamaan (2-11) dan (2-12) diatas dapat dituliskan,

$$K_e(t) - f\dot{\theta}(t) = J\ddot{\theta} + B\dot{\theta} \dots\dots\dots (2-13)$$

Mengambil transformasi laplace di kedua sisi, kita memiliki

$$\frac{\theta(s)}{E(s)} = \frac{K}{Js^2 + (B+f)s} = \frac{K}{s(Js + B+f)} = \frac{K}{s(t_a s + 1)} \dots\dots\dots (2-14)$$

Dimana $Ka = \frac{K}{B+f}$ dan $t_a = \frac{J}{B+f}$ adalah *gain* dan konstanta waktu masing-masing.

Disini, motor servo mengendalikan posisi pembukaan *gate* sesuai perubahan kecepatan pada poros generator untuk mempertahankan kecepatan / frekuensi konstan. Disini, perubahan kecepatan generator berperan sebagai sinyal kontrol.

Persamaan (2-14) adalah fungsi transfer yang dibutuhkan untuk mengembangkan diagram blok lengkap sistem servo hidrolik elektrik.

2.3.2 *Electronic Load Controller (ELC)*

Electronic Load Controller (ELC) merupakan suatu alat kontrol dipergunakan untuk membuang kelebihan daya listrik yang dihasilkan oleh generator pada PLTMH dengan cara mengalihkan beban yang tidak terpakai dengan pengganti (*ballast load*), sehingga frekuensi (Hz), tegangan (volt) serta putaran generator tetap terkendali dan stabil.

ELC telah dikembangkan dengan beberapa metode, dimana untuk mengalihkan kelebihan beban ke *ballast load*, menjaga kestabilan antara masukan *hydro turbine* dan keluaran generator. Dengan variasi beban pengguna, kontroler beban harus mengubah efektifitas resistansi dari beban *ballast*. Maka,

$$P_G = P_C + P_D \dots\dots\dots (2-15)$$

Dimana,

P_G = daya yang dihasilkan pada generator sinkron (harus tetap konstan)

P_C = daya beban pengguna

P_D = daya beban dummy

Daya yang ada di *dump load* dapat digunakan untuk charging baterai, pemanas air, memasak, dan lain-lain.

ELC bekerja berdasarkan sinyal frekuensi dan bekerja untuk mengontrol frekuensi. Rangkaian mainboard akan terus mengecek setiap perubahan frekuensi pada tiap setengah gelombang (10 ms) dan akan menyulut thyristor dan *ballast load* sesuai dengan frekuensi error yang terukur. Penyulutan thyristor dilakukan dengan mengatur sudut fasa sampai frekuensi set point tercapai (50 Hz). Frekuensi akan terus dijaga pada set point 50 Hz dengan akurasi $\pm 0.1-0.2$ Hz dengan *recovery time* sebesar ± 2 detik.

Recovery time merupakan waktu yang dibutuhkan oleh sistem untuk kembali ke *set point* awal ketika terjadi perubahan beban. Respon kontroler dapat diatur melalui dip switches dengan mengatur parameter *Proportional Integral Derivative*

untuk mengatur kecepatan respon kontrol apakah lambat, sedang atau cepat disesuaikan dengan karakteristik beban dan sistem pembangkit.

a. Rectifier

Rectifier adalah rangkaian untuk merubah tegangan AC menjadi tegangan DC. Komponen *rectifier* dapat berupa SCR, MOSFET dan IGBT yang berperan sebagai sakelar. Pada Gambar 2.2 merupakan rangkaian penyearah tiga fasa. Rectifier terdiri dari 5 saklar, S1 dan S4, S3 dan S6, S5 dan S2. Masing-masing saklar S1 dan S4, atau S3 dan S6 atau S5 dan S2, tidak boleh aktif bersamaan, karena bisa menyebabkan *short* pada rangkaian. Pada kondisi aktif dan tidak aktif kedua sisi saklar ditentukan dengan teknik modulasi. Teknik penyalan menggunakan PWM.

Konverter 3 fasa ini telah banyak digunakan dalam industri dengan daya hingga 120 kW. Thyristor dinyalakan dalam interval $\pi/3$. Frekuensi tegangan ripple output adalah 6f, dan membutuhkan penyaringan lebih kecil daripada semikonverter 3 fasa. Selama interval $(\pi/6 + \alpha) \leq \omega t \leq (\pi/2 + \alpha)$, thyristor T_1 & T_6 konduksi dan tegangan line-to-line, $v_{ab} = (v_{an} - v_{bn})$ melalui beban. Jika thyristor-thyristor dalam Gambar 2.2 dinyalakan secara berurutan 12, 23, 34, 45, 56, 61, maka bentuk gelombang yang dihasilkan ditunjukkan dalam Gambar 2.14(b) adalah tegangan input, tegangan output, arus input, dan arus yang melalui thyristor.

$$v_{an} = V_m \sin \omega t \quad (2-16)$$

$$v_{bn} = V_m \sin \left(\omega t - \frac{2\pi}{3} \right) \quad (2-17)$$

$$v_{cn} = V_m \sin \left(\omega t + \frac{2\pi}{3} \right) \quad (2-18)$$

Tegangan line-to-line adalah

$$v_{ab} = v_{an} - v_{bn} = \sqrt{3}V_m \sin \left(\omega t + \frac{\pi}{6} \right) \quad (2-19)$$

$$v_{bc} = v_{bn} - v_{cn} = \sqrt{3}V_m \sin\left(\omega t - \frac{\pi}{2}\right) \quad (2-20)$$

$$v_{ca} = v_{cn} - v_{an} = \sqrt{3}V_m \sin\left(\omega t + \frac{\pi}{2}\right) \quad (2-21)$$

$$V_{dc} = \frac{3\sqrt{3}V_m}{\pi} \cos\alpha \quad (2-22)$$

Tegangan output rata-rata maksimum untuk $\alpha = 0$ adalah

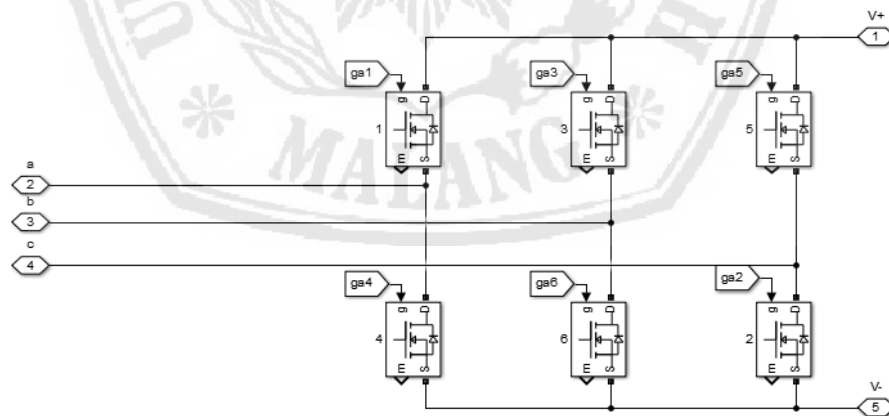
$$V_{dm} = (3\sqrt{3}V_m) / \pi$$

Tegangan output rata-rata dinormalisasi adalah

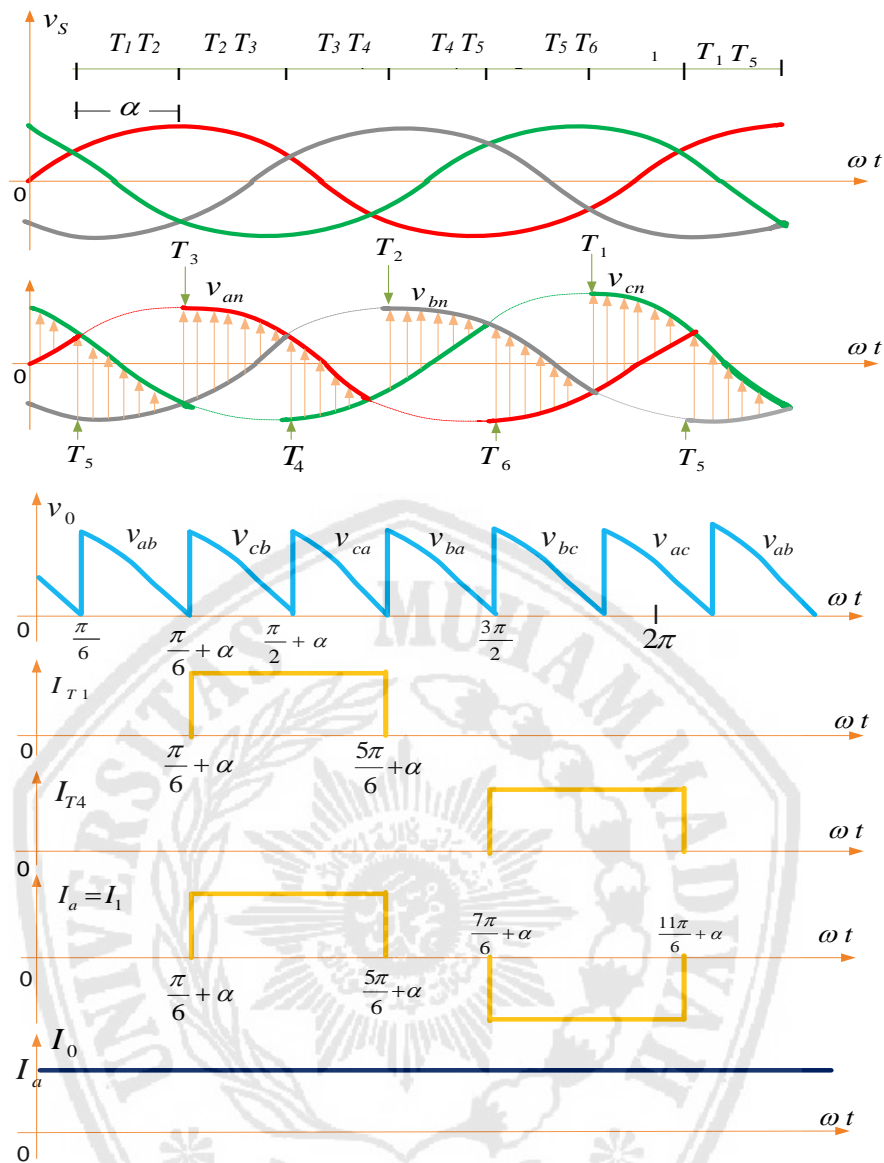
$$V_n = \frac{V_{dc}}{V_{dm}} = \cos\alpha \quad (2-23)$$

Tegangan output rms adalah

$$V_{rms} = \sqrt{6}V_m \left(\frac{1}{4} + \frac{3\sqrt{3}}{8\pi} \cos 2\alpha \right)^{1/2} \quad (2-24)$$



Gambar 2.2 Rangkaian *Rectifier* Tiga Fasa



Gambar 2.3 Bentuk Gelombang *Rectifier*

Tabel 2.1 Urutan Mode Penyalan *Rectifier* 3 Fasa

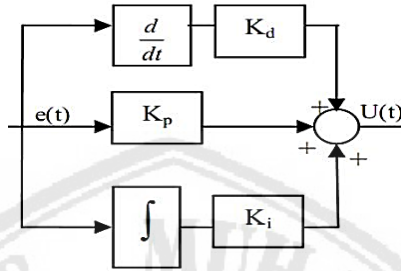
Mode	I	II	III	IV	V	VI
Mosfet	12	23	34	45	56	61
Tegangan	Vac	Vbc	Vab	Vac	Vbc	Vab

2.3.3 Kontroler PID

Kontroler PID adalah kombinasi karakteristik kontrol Proportional (P), Integral (I) dan Derivative (D). Dibawah ini merupakan rumus dasar untuk kontroler PID.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (2-25)$$

Blok diagram kontroler PID ditunjukkan pada Gambar 2.3



Gambar 2.4 Blok Diagram Kontroler PID

Proportional (P) menghasilkan nilai *output* yang sebanding dengan nilai kesalahan saat ini. Respon *proportional* dapat disesuaikan dengan mengalikan kesalahan dengan konstanta K_p , yang disebut konstanta *gain proportional*. Berikut adalah rumus untuk *proportional* :

$$P_{out} = K_p e(t) \quad (2-26)$$

Suatu hasil *proportional* yang tinggi menghasilkan perubahan besar dalam *output* untuk suatu perubahan yang diberikan kedalam nilai *error*. Jika *gain proportional* terlalu tinggi, sistem bisa menjadi tidak stabil. Sebaliknya, hasil yang kecil menghasilkan respons *output* yang kecil terhadap kesalahan *input* yang besar, dan pengontrol yang kurang responsif atau kurang sensitif. Jika perolehan *proportional* terlalu rendah, tindakan kontrol mungkin terlalu kecil saat merespon gangguan sistem. Teori *tuning* dan praktik industri menunjukkan bahwa istilah *proportional* harus menyumbang sebagian besar perubahan *output*.

Integral (I) sebanding dengan besarnya kesalahan dan durasi kesalahan. Integral dalam kontroler PID adalah jumlah kesalahan sesaat dari waktu ke waktu dan memberikan akumulasi *offset* yang seharusnya sudah dikoreksi sebelumnya. Kesalahan yang terakumulasi kemudian dikalikan dengan *gain* integral K_i dan ditambahkan ke output pengontrol.

$$I_{out} = Ki \int_0^t e(\tau) d\tau \quad (2-27)$$

Integral mempercepat gerakan proses menuju *set-point* dan menghilangkan sisa kesalahan *steady state* yang terjadi dengan pengendali *proportional* murni. Namun, karena istilah integral menanggapi akumulasi kesalahan dari masa lalu, itu dapat menyebabkan nilai sekarang untuk melampaui nilai *set-point*

Derivative (D) adalah turunan dari kesalahan proses dihitung dengan menentukan kemiringan kesalahan dari waktu ke waktu dan mengalikan tingkat perubahan ini dengan keuntungan turunan *Kd*. Besarnya kontribusi istilah derivatif terhadap tindakan kontrol secara keseluruhan disebut keuntungan turunan, *Kd*. Istilah derivatif diberikan rumus :

$$D_{out} = Kd \frac{d}{dt} e(t) \quad (2-28)$$

Derivatif meramalkan perilaku sistem dan dengan demikian meningkatkan waktu dan stabilitas sistem. Turunan ideal tidak bersifat kausal, sehingga implementasi pengendali PID mencakup tambahan *low pass filtering* untuk istilah turunan, untuk membatasi perolehan dan kebisingan frekuensi tinggi. Tindakan derivatif jarang digunakan dalam praktik meskipun - dengan satu perkiraan hanya 20% dari pengendali yang dikerahkan - karena dampak variabelnya terhadap stabilitas sistem dalam aplikasi dunia nyata.

2.3.4 Kontroler PI

Kontroler *PI* (*Proportional Integral*) memiliki rumus dasar sebagai berikut:

$$PI(s) = K_p + \frac{K_i}{s} \quad (2-29)$$

Proportional (P) menghasilkan nilai *output* yang sebanding dengan nilai kesalahan saat ini. Respon *proportional* dapat disesuaikan dengan mengalikan kesalahan dengan konstanta *Kp*, yang disebut konstanta *gain proportional*. Berikut adalah rumus untuk *proportional* :

$$P_{out} = K_p e(t) \quad (2-30)$$

Suatu hasil *proportional* yang tinggi menghasilkan perubahan besar dalam *output* untuk suatu perubahan yang diberikan kedalam nilai *error*. Jika *gain proportional* terlalu tinggi, sistem bisa menjadi tidak stabil. Sebaliknya, hasil yang

kecil menghasilkan respons *output* yang kecil terhadap kesalahan *input* yang besar, dan pengontrol yang kurang responsif atau kurang sensitif. Jika perolehan *proportional* terlalu rendah, tindakan kontrol mungkin terlalu kecil saat merespon gangguan sistem. Teori *tuning* dan praktik industri menunjukkan bahwa istilah *proportional* harus menyumbang sebagian besar perubahan *output*.

Integral (I) sebanding dengan besarnya kesalahan dan durasi kesalahan. Integral dalam kontroler PID adalah jumlah kesalahan sesaat dari waktu ke waktu dan memberikan akumulasi *offset* yang seharusnya sudah dikoreksi sebelumnya. Kesalahan yang terakumulasi kemudian dikalikan dengan *gain* integral ***K_i*** dan ditambahkan ke output pengontrol.

$$I_{out} = K_i \int_0^t e(\tau) d\tau \quad (2-31)$$

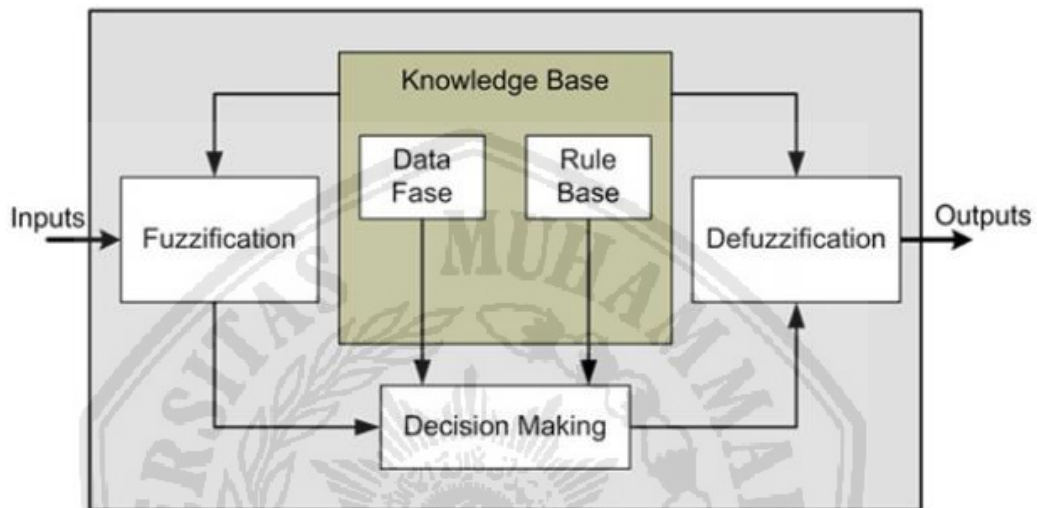
Integral mempercepat gerakan proses menuju *set-point* dan menghilangkan sisa kesalahan *steady state* yang terjadi dengan pengendali *proportional* murni. Namun, karena istilah integral menanggapi akumulasi kesalahan dari masa lalu, itu dapat menyebabkan nilai sekarang untuk melampaui nilai *set-point*.

2.3.5 Adaptive Neuro Fuzzy Inference System (ANFIS)

Memodifikasi fuzzy inference berbasis jaringan (ANFIS) adalah kombinasi dari dua metode *soft-computing* ANN dan logika *fuzzy*. Logika *fuzzy* memiliki kemampuan untuk mengubah aspek kualitatif pengetahuan dan wawasan manusia ke dalam proses analisis kuantitatif yang tepat. Namun, tidak memiliki metode yang ditetapkan yang dapat digunakan sebagai panduan dalam proses transformasi dan pemikiran manusia ke dalam sistem *fuzzy inference rule* berbasis aturan (FIS), dan juga membutuhkan waktu yang cukup lama untuk menyesuaikan fungsi keanggotaan (MFs). Tidak seperti JST, ia memiliki kemampuan yang lebih tinggi dalam proses pembelajaran untuk beradaptasi dengan lingkungannya. Oleh karena itu, JST dapat digunakan untuk secara otomatis menyesuaikan MF dan mengurangi tingkat kesalahan dalam penentuan aturan dalam logika fuzzy. Bagian ini akan menjelaskan secara detail arsitektur ANFIS, FIS, dan fleksibilitas jaringan, serta algoritma pembelajaran hibrid.

2.3.5.1 Fuzzy Inference System

FIS dibangun pada tiga komponen utama, yaitu aturan dasar, di mana ia terdiri dari pemilihan aturan logika fuzzy "*If-Then*;" sebagai fungsi dari keanggotaan set *fuzzy* dan penalaran teknik inferensi *fuzzy* dari aturan dasar untuk mendapatkan *output*. Gambar 2.4 menunjukkan struktur terperinci FIS.



Gambar 2.5 Fuzzy inference system

FIS akan bekerja ketika input yang berisi nilai aktual diubah menjadi nilai fuzzy menggunakan proses fuzzification melalui fungsi keanggotaannya, di mana nilai fuzzy memiliki rentang antara 0 dan 1. Aturan dasar dan basis data disebut sebagai basis pengetahuan, di mana keduanya merupakan elemen kunci dalam pengambilan keputusan. Biasanya, database berisi definisi seperti informasi pada parameter set *fuzzy* dengan fungsi yang telah ditetapkan untuk setiap variabel linguistik yang ada.

Pengembangan database biasanya mencakup mendefinisikan alam semesta, penentuan jumlah nilai linguistik yang akan digunakan untuk setiap variabel linguistik, serta menetapkan fungsi keanggotaan. Berdasarkan aturan, ini berisi operator logika fuzzy dan pernyataan bersyarat "*If-Then*." Aturan dasar dapat dibangun baik dari generasi manusia atau otomatis, di mana aturan pencarian menggunakan input-output data

secara numerik. Ada beberapa jenis FIS, yaitu Takagi – Sugeno, Mamdani, dan Tsukamoto. Model FIS Takagi Sugeno ditemukan banyak digunakan dalam aplikasi metode ANFIS.

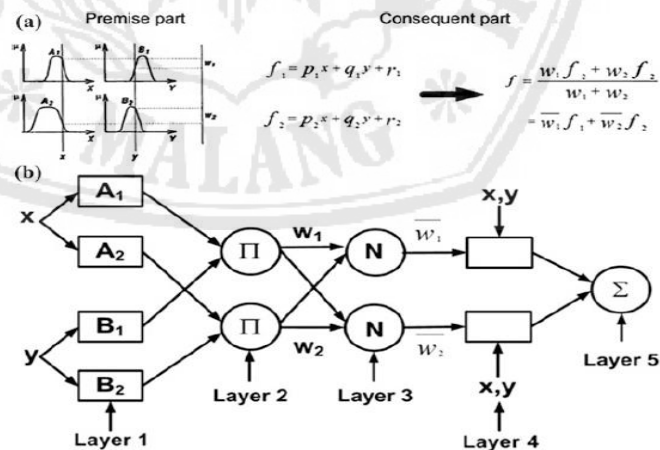
2.3.5.2 Arsitektur ANFIS

Arsitektur ANFIS adalah jaringan adaptif yang menggunakan pembelajaran yang diawasi pada algoritma pembelajaran, yang memiliki fungsi yang mirip dengan model sistem inferensi *fuzzy* Takagi – Sugeno. Gambar 2.5a, b menunjukkan skema mekanisme penalaran *fuzzy* untuk model Takagi – Sugeno dan arsitektur ANFIS. Untuk kesederhanaan, asumsikan bahwa ada dua input x dan y , dan satu output f . Dua aturan digunakan dalam metode "If-Then" untuk model Takagi – Sugeno, sebagai berikut:

Rule 1 = If x is A_1 and y is B_1 Then $f_1 = p_1x + q_1y + r_1$

Rule 2 = If x is A_2 and y is B_2 Then $f_2 = p_2x + q_2y + r_2$

Dimana A_1, A_2 dan B_1, B_2 adalah fungsi keanggotaan dari setiap input x dan y (bagian dari tempat), sedangkan p_1, q_1, r_1 dan p_2, q_2, r_2 adalah parameter linear di bagian-Kemudian (bagian konsekuen) dari Takagi - Sugeno *fuzzy inference model*.



Gambar 2.6 (a) Sugeno Fuzzy inference system (b) ANFIS Arsitektur

Layer 1: Setiap node dalam layer ini beradaptasi dengan parameter fungsi. Output dari setiap node adalah tingkat nilai keanggotaan yang

diberikan oleh input dari fungsi keanggotaan. Sebagai contoh, fungsi keanggotaan dapat menjadi fungsi keanggotaan Gaussian, fungsi keanggotaan umum, atau jenis fungsi keanggotaan lainnya.

$$\mu_{Ai}(x) = \exp \left[- \left(\frac{x - c_i}{2a_i} \right)^2 \right] \quad (2-32)$$

$$\mu_{Ai}(x) = \frac{1}{1 + \left| \frac{x - c_i}{a_i} \right|^{2b}} \quad (2-33)$$

Dimana μ_{Ai} dan μ_{Bi} adalah tingkat fungsi keanggotaan untuk fuzzy menetapkan A_i dan B_i , masing-masing, dan $\{a_i, b_i, c_i\}$ adalah parameter dari fungsi keanggotaan yang dapat mengubah bentuk fungsi keanggotaan. Parameter dalam lapisan ini biasanya disebut sebagai parameter premis.

Layer 2: Setiap node dalam lapisan ini tetap atau non-adaptif, dan simpul lingkaran diberi label sebagai P. *Output* node adalah hasil penggandaan sinyal yang masuk ke-node dan dikirimkan ke-node berikutnya. Setiap node dalam layer ini mewakili kekuatan tembak untuk setiap aturan. Pada lapisan kedua, operator T-norm dengan kinerja umum, seperti AND, diterapkan untuk mendapatkan output.

$$O2_i = w_i = \mu_{Ai}(x) * \mu_{Bi}(y), \quad i = 1, 2 \quad (2-34)$$

Dimana w_i adalah *output* yang mewakili kekuatan *firing* dari setiap aturan.

Layer 3: Setiap node dalam lapisan ini tetap atau non-adaptif dan simpul lingkaran diberi label sebagai N. Setiap node adalah penghitungan rasio antara kekuatan pengaktifan aturan ke- i dan jumlah dari semua kekuatan pengaktifan aturan. Hasil ini dikenal sebagai kekuatan penembakan yang dinormalkan.

$$O3_i = \overline{w}_i = \frac{w_i}{\sum_i w_i} \quad (2-35)$$

Layer 4: Setiap node dalam layer ini adalah node adaptif ke output, dengan fungsi node didefinisikan sebagai,

$$O4_i = \overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i) \quad (2-36)$$

Dimana w_i adalah kekuatan menembak yang dinormalisasi dari lapisan sebelumnya (lapisan ketiga) dan $(p_i x + q_i y + r_i)$ adalah parameter

dalam node. Parameter dalam lapisan ini disebut sebagai parameter konsekuen.

Layer 5: Simpul tunggal dalam lapisan ini adalah node tetap atau non-adaptif yang menghitung keseluruhan output sebagai penjumlahan semua sinyal yang masuk dari node sebelumnya. Dalam lapisan ini, simpul lingkaran diberi label sebagai Σ

2.3.6 Fuzzy Logic Controller

Fuzzy Logic ditemukan dan digunakan oleh Prof. Lotfi Zadeh, tahun 1965. Kontrol fuzzy memberikan metodologi formal untuk mewakili, memanipulasi, dan menerapkan pengetahuan heuristik manusia tentang cara mengontrol sistem. Fuzzy merupakan pembuat keputusan buatan yang beroperasi dalam sistem loop tertutup secara real-time.

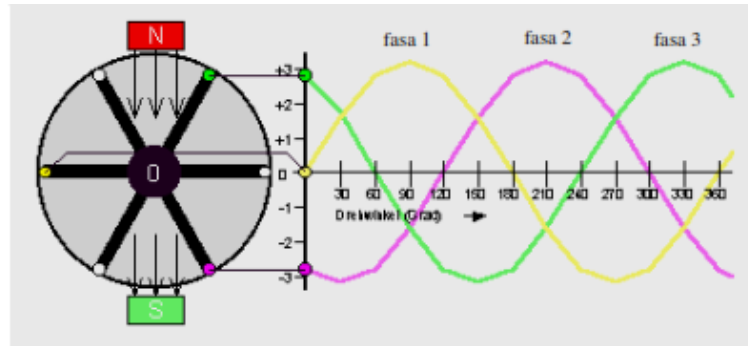
Fuzzy mempunyai dua input antara lain nilai error dan nilai turunan pertama dari nilai error. Untuk mendesain fuzzy perlu mendesain fungsi keanggotaan, aturan IF-THEN dan basis pengetahuan. Fuzzy mempunyai urutan dalam pembuatan sistemnya, antara lain:

1. Fuzzification Process.
2. Inference Mechanism Process.
3. Defuzzification Process.

2.3.7 Generator Sinkron

Generator sinkron adalah jantung dari energi listrik dalam sistem tenaga listrik. Studi tentang teori dan pertunjukan mesin sinkron pada akhirnya adalah pemahaman konsep stabilitas sistem tenaga. Mesin sinkron dapat diklasifikasikan dalam dua kategori. Generator berkecepatan tinggi yang dikenal sebagai generator turbo biasanya digerakkan oleh gas, diesel, steam dll, dan generator berkecepatan rendah yang digerakkan oleh sistem hidrolik. Sirkuit *armature* generator sinkron ditempatkan pada stator dan gulungan medan dan gulungan damper ditempatkan pada rotor sepanjang sumbu-d dan sumbu-q. Gulungan membawa arus beban dan memasok daya ke sistem. Sirkuit eksitasi generator sinkron biasanya

mengembangkan fluks yang berputar sehingga angker dapat merespon dengan memanfaatkan fluks ini.



Gambar 2.7 Pembangkit tegangan 3 fasa

Frekuensi yang dihasilkan oleh *synchronous generator* adalah menyamakan dengan kecepatan putar generator. *Rotor synchronous generator* terdiri dari rangkaian elektromagnet dengan menyuuplai arus DC. Medan magnet pada *rotor* berputar pada arah putaran *rotor*. Dapat dilihat pada persamaan berikut :

$$f = \frac{n \cdot P}{120} \dots\dots\dots(2-37)$$

dimana :

f = electric frequency [Hz]

n = magnetic field speed = rotor speed [rpmm]

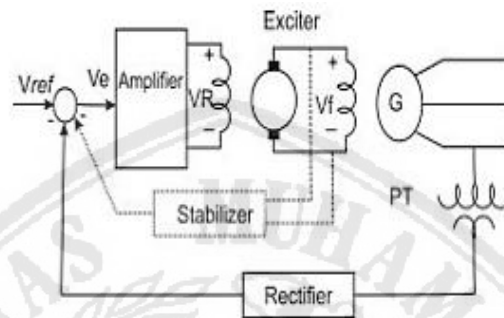
P = the number of poles

2.3.8 Eksitasi

Excitasi adalah sumber daya listrik untuk gulungan medan generator dan direalisasikan sebagai generator DC atau AC terpisah. Exciter memiliki medan berliku di stator, dan belitan *armature* di rotor. Dalam kasus generator AC, ketika rotor berputar, arus stator DC menginduksi arus bolak-balik tiga fasa ke dalam rotor berliku. Arus AC ini diperbaiki menggunakan dioda, thyristor atau jembatan transistor yang dipasang di rotor.

Exciter dikendalikan oleh AVR, yang sangat efektif selama *operasi steady-state*, dalam kasus gangguan tiba-tiba itu mungkin memiliki pengaruh negatif pada redaman perubahan kekuatan, karena kemudian memaksa perubahan medan saat ini di generator. Ini dapat dihilangkan dengan memperkenalkan loop kontrol tambahan,

penstabil sistem daya (PSS), yang menghasilkan sinyal tambahan ke dalam lingkaran kontrol dan dengan cara itu mengkompensasi osilasi tegangan. Rentang frekuensi osilasi yang khas dan tidak cukupnya redaman osilasi ini dapat membatasi kemampuan untuk mentransmisikan daya. Jumlah input PSS dapat berupa kecepatan penyimpangan, daya aktif generator, penyimpangan frekuensi, gaya gerak listrik sementara dan arus generator. Biasanya dua dari jumlah input ini adalah chosento mendapatkan regulasi optimal.



Gambar 2.8 Model Eksitasi

2.3.8.1 Amplifier

Amplifier merupakan *gain* dari eksitasi dapat berupa *gain magnetic*, *rotor gain*, atau *electroic gain*. *Amplifier* ditunjukkan K_A dengan *time constant* T_A , yang dalam model matematisnya seperti persamaan berikut.

$$\frac{V_R(s)}{V_s(s)} = \frac{K_A}{1+T_A s} \dots\dots\dots(2-38)$$

2.3.8.2 Voltage Regulator

Sistem eksitasi stabilisasi sering dicapai dalam sistem *thyristor* oleh jaringan lag lead seri daripada melalui umpan balik tingkat. Waktu konstanta, T_B dan T_C .

$$\frac{V_R(s)}{V_s(s)} = \frac{1+T_C s}{1+T_B s} \dots\dots\dots(2-39)$$

2.3.8.3 Exciter

Eksitasi yang digunakan pada generator memiliki beberapa tipe, pertama menggunakan generator DC sampai yang tipe modern dengan menggunakan SCR sebagai penyearah sebagai penghasil daya DC. Eksitasi

dapat dipresentasikan sebagai K_E dengan konstanta waktu T_E , yang dalam model matematisnya seperti persamaan berikut:

$$\frac{V_F(s)}{V_R(s)} = \frac{K_E}{1+T_E s} \dots\dots\dots(2-40)$$

2.3.8.4 Sensor

Sensor terdiri atas transformator tegangan dan sebuah penyearah. Sensor dapat dipresentasikan sebagai K_R dengan konstanta waktu T_R dan fungsi transfernya sebagai berikut:

$$\frac{V_S(s)}{V_t(s)} = \frac{K_R}{1+T_R s} \dots\dots\dots(2-41)$$

Prinsip kerja dari eksitasi yaitu sebagai tegangan keluaran generator mulanya diturunkan dengan menggunakan PT (*Potential Transformer*) atau trafo tegangan kemudian disearahkan. Hasil penyearahan lalu dibandingkan dengan tegangan referensi (V_{ref}) apabila terjadi perbedaan maka eksitasi akan memerintahkan amplifier untuk menaikkan atau menurunkan arus penguatan generator DC sehingga tegangan *output* dari generator tersebut juga berubah. Jika tegangan *output* generator DC berubah maka arus penguatan generator sinkron juga berubah, akibatnya tegangan keluaran generator kembali stabil.